**Predictia nivelului de lumina dintr-o incapere folosind Descompunerea STL**

**Achizitia setului de date**

Ca set de date, am ales sa culeg informatii legate de luminozitatea dintr-o camera pe parcursul mai multor zile. In acest sens, am folosit un microcontroller din familia ESP32 si un fotorezistor. Avand un modul WiFi integrat, acesta poate transmite date catre platforma InitialState, care mai apoi le transforma in grafice si le salveaza intr-un fisier pe care il vom folosi in proiectul nostru. Codul scris in Arduino IDE consta in conectarea la reteaua de internet, compunerea url-ului ce contine 2 tipuri de chei asociate cu stream-ul de date din contul meu si valoarea luminozitatii insotita de numele semnalului sub forma cheie-valoare. Acest URL este transmis unui server si asteapta un mesaj de confirmare (un cod 204). Codul va fi inclus in arhiva.

O imagine care conține electronice, Inginerie electronică, circuit, Componentă electronică

Descriere generată automat

Masuratorile sunt efectuate o data la 6 minute, timpul de transmisie variind cu cateva secunde in functie de calitatea conexiunii. Un circuit asemanator poate fi regasit in stanga.

Fisierul *data.csv* folosit in program are structura de mai jos:

*2023-12-03T07:49:19.709545000Z,light,60*

*2023-12-03T07:43:17.613781000Z,light,14*

*2023-12-03T07:37:15.635863000Z,light,48*

Tabelul are pe prima coloana momentul de timp, pe a doua, numele semnalului iar pe ultima, valoarea efectiva.

Ne-am astepta ca structura datelor sa fie una relativ omogena: parabole intercalate cu portiuni unde luminozitatea este 0. Insa, multe din zilele cand s-au efectuat masuratorile au fost ploioase, fapt ce se reflecta in setul de date. De asemenea, sunt cateva valori aberante care vor fi eliminate in etapa de preprocesare (momente cand cineva aprinde din greseala lumina pe parcusul noptii/ acopera senzorul pe timp de zi). Trendul se pastreza totusi iar la intervale de timp aproximativ egale luminozitatea creste/scade gradual, fapt ce face posibila o eventuala predictie.

**Preprocesarea datelor**

In privinta preprocesarii, am ales un algoritm prezentat in curs, si anume, filtrul median cu o fereastra de 3 unitati. Il consider potrivit, deoarece abaterile sunt punctuale, si este mai potrivit sa inlocuim valoarea afectata cu o valoare vecina, decat sa facem, spre exemplu, o medie intre ele. Fereastra de 3 a fost aleasa deoarece sunt sanse mai mari ca luminozitatea reala sa fie egala cu una masurata cu 6 minute inainte sau dupa, decat cu una masurata cu 12 sau 18 minute inainte sau dupa. Rezultatele filtrarii poti fi vazute mai jos:

O imagine care conține text, captură de ecran, diagramă, Interval

Descriere generată automat

O imagine care conține captură de ecran, Interval, linie

Descriere generată automat

**Algoritmi utilizati**

In arhiva proiectului sunt 3 programe + fisierul data.csv + last, un fisier ce contine ultimele 2 zile:

1. In main.py, avem algoritmul din laboratorul 3 folosit pe setul de date ca punct de pornire. Prezice urmatoarele 10 valori pentru seria de timp data.
2. In algoritm1.py am folosit un algoritm foarte simplu, Moving Average, care calculeaza media dintre valorile regasite intr-o fereastra de dimensiune setata manual. Nu este un algoritm foarte util in cazul nostru.

O imagine care conține text, captură de ecran, diagramă, Interval

Descriere generată automat

1. Intrucat seria noastra este una periodica, putem cauta un algoritm dedicat acestor tipuri de serii. Am folosit *STL Decomposition*- un algoritm ce evalueaza o serie de timp si o imparte in 3 componente: seasonal, trend, residual. Dupa ce am antrenat modelul pe cateva zile de date, am extras partea seasonal si trend, deoarece am considerat ca partea reziduala nu este importanta si trebuie eliminata iar apoi am afisat valorile prezise de modelul meu pentru cateva perioade.

Perioada: reprezinta numarul de masuratori dupa care ciclul se reia. In cazul nostru, avem 10 masuratori pe ora timp de 24h => perioada= 240. Ca grafice afisate, avem rezultatul preprocesarii, datele+predictia, componentele datelor, comparatie realitate-predictie.

O imagine care conține text, captură de ecran, diagramă, Interval

Descriere generată automat

O imagine care conține text, captură de ecran, Interval, diagramă

Descriere generată automatO imagine care conține text, linie, Interval, diagramă

Descriere generată automat

O imagine care conține captură de ecran, Interval, text, diagramă

Descriere generată automatO imagine care conține text, Font, captură de ecran, Grafică

Descriere generată automat

**Evaluarea modelului**

Aceasta poate fi facuta in primul rand, vizual, in acest sens am afisat ultimele doua zile de masuratori(fara preprocesare) si predictiile pentru 2 zile. Am calculat MSE, valoarea ei fiind destul de mare, deoarece, daca ne uitam la ultimele zile, acestea au fost cele mai putin luminoase din saptamana, deci algoritmul nostru prezice o cantitate mai mare de lumina. La nivel de trend si perioada, insa, este satisfacator.

**Biblioteci folosite:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import csv  
from statsmodels.tsa.seasonal import STL  
from scipy.signal import medfilt  
from sklearn.metrics import mean\_squared\_error